



## Aircraft trajectory forecasting using local functional regression in Sobolev space

Kairat Tastambekov, Stéphane Puechmorel, Daniel Delahaye, Christophe Rabut

### ► To cite this version:

Kairat Tastambekov, Stéphane Puechmorel, Daniel Delahaye, Christophe Rabut. Aircraft trajectory forecasting using local functional regression in Sobolev space. Transportation research. Part C, Emerging technologies, 2014, 39, pp 1-22. 10.1016/j.trc.2013.11.013 . hal-00924360

**HAL Id: hal-00924360**

**<https://hal-enac.archives-ouvertes.fr/hal-00924360>**

Submitted on 5 Feb 2014

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Aircraft Trajectory Forecasting using Local Functional Regression in Sobolev Space

K. Tastambekov<sup>a</sup>, S. Puechmorel<sup>a</sup>, D. Delahaye<sup>a</sup>, C. Rabut<sup>b</sup>

*<sup>a</sup>Ecole National de l'Aviation Civile, Département de Mathématiques Appliquées, Informatique et Automatique pour l'Aérien,*

*7, Avenue Edouard Belin, 31055 Toulouse, France*

*<sup>b</sup>Institut National des Sciences Appliquées, Département de Génie Mathématique et Modélisation,*

*135, Avenue de Rangueil, 31077 Toulouse, France*

---

---

*Email addresses:* kairat@recherche.enac.fr (K. Tastambekov),  
puechmor@recherche.enac.fr (S. Puechmorel), delahaye@recherche.enac.fr  
(D. Delahaye), christophe.rabut@insa-toulouse.fr (C. Rabut)

## **Abstract**

This paper considers the problem of short to mid-term aircraft trajectory prediction, that is, the estimation of where an aircraft will be located over a 10 to 30 minutes time horizon. Such a problem is central in decision support tools, especially in conflict detection and resolution algorithms. It also appears when an air traffic controller observes traffic on the radar screen and tries to identify convergent aircraft, which may be in conflict in the near future. An innovative approach for aircraft trajectory prediction is presented in this paper. This approach is based on local linear functional regression that considers data preprocessing, localizing and solving linear regression using wavelet decomposition. This algorithm takes into account only past radar tracks, and does not use any physical or aeronautical parameters. This approach has been successfully applied to aircraft trajectories between several airports on the data set that is one year air traffic over France. The method is intrinsic and independent from airspace structure.

**Keywords:** Trajectory prediction, functional regression, Air Traffic Management

## Nomenclature

|  |   |  |
|--|---|--|
| $\{\Omega, \mathfrak{F}, \mathbb{N}\}$ | = | Probability space                                      |
| $X_\omega(t), Y_\omega(s)$             | = | Hilbert random processes                               |
| $X_n$                                  | = | predictor variable coming from $X_\omega(t)$           |
| $Y_n$                                  | = | response variable coming from $Y_\omega(s)$            |
| $\tau_X^n, \tau_Y^n$                   | = | time intervals related to $X_n$ and $Y_n$ respectively |
| $W^k(\mathbb{R})$                      | = | Sobolev space of order $k$                             |
| $\lambda_n$                            | = | local weight   |
| $\Psi^x, \Psi^y$                       | = | Two different wavelet basis used for expansions        |
| $a_{n,i}, b_{n,j}, c_{i,j}$            | = | wavelet coefficients                                   |
| $\{V\}_{j \in \mathbb{Z}}$             | = | linear spaces generated by wavelet functions           |
| $\beta(t, s)$                          | = | square-integrable matrix-valued function               |
| $t, s$                                 | = | time   |
| $i, j, k, n$                           | = | waypoint   |

## List of Figures

|    |  |    |
|----|--|----|
| 1  | Training set for multiple linear regression. . . . .   | 48 |
| 2  | Altitude trajectory prediction (dashdot line) and the original path<br>(line). . . . .   | 49 |
| 3  | Predicted trajectory (dashdot line) and real trajectory (line). . . . .  | 50 |
| 4  | Example of a real trajectory and the corresponding prediction (left)<br>and real trajectory not included in the learning base and corre-<br>sponding prediction (right). . . . . | 51 |
| 5  | Example of altitude of 20 aircraft trajectories during take-off. . . . .   | 52 |
| 6  | 23 aircraft trajectories (altitude) during take-off, cruising and land-<br>ing. . . . .  | 53 |
| 7  | Mean of the 23 aircraft trajectories (altitude) of Figure 6 . . . . .  | 54 |
| 8  | Leave-one-out cross-validation procedure. . . . .  | 55 |
| 9  | Benefits of considering trajectories as functions. . . . .   | 56 |
| 10 | An aircraft's x coordinate and its CWT . . . . .   | 57 |
| 11 | An aircraft's y coordinate and its CWT . . . . .   | 58 |
| 12 | An aircraft's z coordinate and its CWT . . . . .   | 59 |
| 13 | Smoothing and compressing an artificial function with wavelets. . . . .  | 60 |
| 14 | $X \times Y$ plane of one day air traffic covered by ATC radars in France. . . . .   | 61 |
| 15 | Real trajectory (line) and Predicted trajectory (dashdot line). Paris<br>Orly - Nice. . . . .  | 62 |
| 16 | Real trajectory (line) and Predicted trajectory (dashdot line). Paris<br>Orly - Marseille. . . . .   | 63 |
| 17 | Real trajectory (line) and Predicted trajectory (dashdot line). Paris<br>Orly - Toulouse . . . . .   | 64 |

|    |  |    |
|----|--|----|
| 18 | Real trajectory (line) and Predicted trajectory (dashdot line). . . .                          | 65 |
| 19 | Relative prediction errors for 40 trajectories between Paris and<br>Toulouse airports. . . . . | 66 |

## **1. Introduction**

### *1.1. Basics of Air Traffic Control*

Air Traffic Control (ATC) is a service provided by ground-based air traffic controllers, who manage aircraft from departure until arrival. Air Traffic Management (ATM) research and development has produced numerous Decision Support Tools (DST) aimed at providing controllers with automated conflict detection and resolution, trajectory planning, and aids for sequencing arrivals and departures at airports. At a strategic level (that is several months before flights take place), assessing controllers workload is a major concern in order to keep it within acceptable bounds and take mitigating actions if needed.

In many parts of the world, ATM induces a large amount of control workload (for example, about 8000 aircraft per day fly in the French airspace). It is not possible for a single or even a team of controllers to manage such a workload. To cope with this problem, airspace is divided into polygonal cells called sectors. A team of 2 controllers is in charge of a control sector and performs three essential tasks: monitoring (that is checking compliance of the traffic with regulatory separations), conflict resolution, and coordination (which deals with aircraft transfer from and to adjacent sectors. The first and second task heavily rely on the ability to predict where an aircraft will be located in a 10 minutes time window. From this observation, it is clear that controller's workload can be reduced by an accurate and automated forecast of aircraft trajectories, since the knowledge of whether conflicts are to be expected or not helps in prioritize the actions to be taken.

A major concern when dealing with trajectory prediction is the robustness of forecast with respect to unknown or partially known influential parameters. Many different factors have an adverse impact on the accuracy of prediction and depend



on the time horizon for the forecasting. In principle, the knowledge of the flight dynamics equations for a given aircraft, the intended flight plan and exogenous parameters like temperature, wind and ATC orders will be enough to accurately model a trajectory from departure to destination. Unfortunately, many of these factors are unknown or partially known only (as an example, the take-off mass of the aircraft, or the wind during the flight can only be guessed). This is the major limitation of model based predictors, and it explains why they perform quite poorly on a real data).

Another approach is to assume that trajectories are realizations of some random process and regression techniques applied to previous trajectories. Invoking Takens theorem, it can be shown that a sufficient number of samples is enough to reconstruct a dynamics conjugate to the real one based on flight equations and thus that regression algorithms are in principle as powerful as model based predictors. Unfortunately, estimating the size of the sample to be used is a quite difficult challenge, and we cannot oversize the number of samples since having too long sequences has a detrimental effect on the numerical stability of the algorithms. The approach proposed here removes some of the limitations of classical regression by considering trajectories as functions (elements of a Sobolev space), instead of considering temporal samples of them. We also use a convenient basis for representing the so-used functions, and so limit the amount of computation.

## *1.2. Trajectory Prediction in decision support tools*

As mentioned above, the knowledge of aircraft positions along time, termed as the '4D trajectory' in the context of future ATM systems, is a key ingredient for improving the capacity of the airspace and will ultimately allow the system to cope with the anticipated two to three fold increase in the number of flights by

2030.

*Early conflict prediction.* The objective of conflict prediction and resolution algorithms is to anticipate conflict situations between two or more aircraft and to propose mitigating actions to controllers (in the 2050 horizon, full automation is also an option, with the presence of humans only for monitoring purposes). Let us mention however that automated conflict resolution is a complicated task that is not yet fully solved by algorithms. The question of whether a conflict will exist with a high probability has to be answered before triggering the solver. Trajectory prediction (TP) is obviously a key ingredient for the design of efficient automated conflict solver. In such a context, the accuracy requirements stem from the need of keeping the separation in early detection.

The assessment of TP performance will be made using a composite indicator:

- Evaluation of the level of false alarms and non-detection on a test data set;
- When detection is correctly triggered, the performance is the distance in space and time between the forecast point of conflict and the real one.

The conflict prediction has the highest requirements among the two main uses of TP, and is still an open issue. The problem is worsening by the need for wind knowledge along the flight path that presently is unfortunately currently available only through weather forecast services, with a quite coarse precision and resolution both in space and time. This topic is far from being solved yet, and speaks in favor of regression based predictor since wind information can be taken into account from past trajectories observed in the vicinity (in time and space) of the area of interest.

*Landing sequence.* According to EUROCONTROL's statement, the general objective of management software is to assist, through dedicated automation, in the management of the flow of arriving traffic in a particular airspace and to particular points. It aims at optimizing the landing times of arrivals (sequence); regulate and manage the flow of aircraft to improve predictability, reduce workload and at the same time minimize the impact on the environment. Trajectory prediction assists air traffic controllers of the arrival flow, particularly in the case of perturbations (such as runway closure). Here, forecasting of arrival time is the relevant objective. The spatial accuracy is less important since traffic is expected to merge at a given entry point. The wind issue already mentioned has unfortunately the same adverse effect as in the conflict detection problem since flight management systems are not efficient at ensuring low along track error, and wind velocity will have a direct impact on the time needed to reach a given point on the flight plan. The performance assessments of TP on the problem of sequencing will easily be done by comparing a predicted arrival time to the real one, for a global test data set. It is clear that solving the first problem will in turn solve also the sequencing one.

### *1.3. Summary*

Despite the increasing autonomy in the future air traffic management systems, especially 4D trajectory planning, the ability of trajectory prediction is still of high importance. As known, 4D trajectory planning implies aircraft to be properly located in certain place at certain time. But, such an approach is not realistic. Presently, Flight Management Systems are not able to impose an aircraft keep a given path or be at a given point at given time. Here the major factor is the wind. For example, by the official information from Airbus, in 4D trajectory planning

for one hour flight the along path error is about 15 Nautical miles (NM). Thus, trajectory prediction problem can be stated as an crucial issue at least for near future.

Several known methods to solve the problem of trajectory prediction are described in part 2 of the paper (some of them in (C.Gong and D.McNally, 2004), (Y.L.Fablec and Y.Alliot, 1999), (M.Serrurier and Prade, 2004), (H.Choi, August 2010). Part 3 outlines the new approach to dealing with aircraft trajectories, considering them to be functional data. Also, the wavelet theory in common has been described there. The next part introduces the associated mathematical modeling environment and describes the main results of the paper. Finally, the last part shows simulation results and the efficiency of the method on a recorded one year air traffic data over France.

## **2. State-of-the-Art**

### *2.1. Trajectory Prediction based on Multiple Linear Regression*

Firstly, the standard approach consisting of multiple linear regression for aircraft trajectory prediction is applied to the same data set as in the main part of the paper. Typically, standard linear regression can be seen a kind of Kalman filter, since:

- Both Kalman filter and linear regression use measurements observed over time to produce estimates as close as possible to the true value of the estimated variables.
- In both cases the training set contains noise and other uncertainties.

One difference is that the Kalman filter is a recursive method, whereas the linear regression uses all past observation in batch mode to predict the future value of the variable of interest.

The general linear regression model is defined as:

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_m x_m + \varepsilon \quad (1)$$

where

- $y$  is a dependent or response variable
- $x_1, \dots, x_n$  are independent or predictor variables
- $\beta_j$  are parameters
- $\varepsilon$  is an error with zero mean.

The linear least square estimator is written:

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_m x_m \quad (2)$$

where  $\hat{y}$  is the estimate of  $y$  and  $\hat{\beta}_i$  is an estimate of  $\beta_i$ ,  $i = 1, \dots, m$ . Parameters  $\hat{\beta}_0, \dots, \hat{\beta}_m$  can be obtained by minimizing the sum of squared residuals in Linear Least Squares Method, that is, for a given set of  $n$  observations  $\{y_i, x_{i1}, \dots, x_{im}\}$ , the estimated values  $\hat{\beta}_0, \dots, \hat{\beta}_m$  result from minimizing:

$$SSR = \sum_{i=1}^n (y_i - \sum_{j=1}^m \beta_j x_{ij})^2 \quad (3)$$

over the  $\beta_j$ ,  $j = 1, 2, \dots, m$ . In order to simulate the regression we use a set of  $n = 293$  aircraft trajectories. We note the trajectory to be predicted as  $y$  and

training set of predictor variables as  $x_i, i = 1, \dots, m$ . Figure 1 shows full 293 trajectories that are used to build the regression model. From now, the goal is to forecast one of the trajectory from the training set and available trajectory data up to the time at which the forecast is made.

Place figure 1 about training set for multiple linear regression here.

A simulated prediction can be seen from figures 2 and 3. On both figures prediction starts from time  $t = 130$  seconds. The predicted trajectory is shown in dashdot and real trajectory of the aircraft is shown as normal line. The major issue with this regression approach lies with the fact that too many parameters are used as inputs to the regression model: in fact, all observations were used to solve the regression. The result on figure 2 is that the model is overfitted to the data, and it is highly sensitive to noise and variations of the input parameters. On the other hand figure 3 illustrates the behavior of the linear regression method, when the number of observations used in the training set is reduced to one-fourth of the available data set, that is, for each trajectory from the same data set, as in figure 1, 1 out of 4 points is used. It is clearly seen from the figures that too many parameters entering the initial model produces big oscillations in prediction and a poor generalization of the model, but a too small number of parameters can lead to a poor description of the shape of the trajectory (G.Lebanon, 2010). No doubt, data overfitting and model generalization can be stated as the main problem of standard regression method, particularly multiple linear regression models.

Place figures 2 and 3 about altitude trajectory prediction and predicted trajectory here.

In section IV below, the reader will see that we expand our approach by assuming the aircraft trajectories are functional data and by relying on methods

that allow us to identify the optimal number of input parameters while conserving enough information on the shapes of the trajectories. One of such method is called, in generally, Cross-Validation procedure. It also estimates the prediction accuracy of statistical models and assesses how they generalize to an independent data set. As it will be seen in section IV, solving the problem of choosing the optimal number of parameters in the model can reduce the Mean Square Error to the minimum. Before doing so, we turn our attention to other existing methods available from existing literature.

## 2.2. *Using Neural Networks to Predict Aircraft Trajectories*

This section is based on the work of Yann le Fablec and Jean Marc Alliot - "Using Neural Networks to Predict Aircraft Trajectories" (Y.L.Fablec and Y.Alliot, 1999). In this paper, the authors deal with the problem of aircraft trajectory prediction in the vertical plane. They propose two steps: (i) examining the method that depends on a small number of starting parameters; (ii) extend the method on wide range of cases. The chosen method is based on neural networks that are well known for having universal approximation property of continuous mappings (Kolmogorov theorem and variants). Since neural networks are able to approximate arbitrary well any continuous mapping, there are good candidates for improving over the multiple linear regression already described. The standard architecture for such purpose is one hidden layer with sigmoid activation function and an output layer, that is a single linear neuron. Given input  $x_i, i = 1 \dots N$ , the output of the neural network is given by:

$$y = \sum_{j=1 \dots M} a_j f_j \left( \sum_{i=1 \dots N} w_{ji} x_i + b_i \right) + c \quad (4)$$

where the scalars  $w_{ij}$ ,  $b_i$  are respectively the weights and the biases of the hidden neurons, and  $a_j$  and  $c$  the weights and bias of the output neuron. The activation functions  $f_j$  are usually all equal, and in the present application, selected to be the hyperbolic tangent. Finding the parameters  $w_{ij}$ ,  $b_i$ ,  $a_i$ ,  $c$  of the neural network is obtained by minimizing the prediction error over the learning database: given samples  $(x_{1k}, \dots, x_{Nk}, y_k)$ ,  $k = 1 \dots M$ , one has to solve the least mean square problem:

$$\min_{w_{ij}, b_i, a_i, c} \sum_{k=1}^M \left( y_k - \sum_{j=1 \dots M} a_j f_j \left( \sum_{i=1^N} w_{ji} x_{ik} + b_i \right) + c \right)^2 \quad (5)$$

A popular algorithm for finding the optimal parameter is the steepest descent, efficiently computed by the so-called back propagation of the gradient. As usual with this kind of method, quite a huge number of iterations are needed to reach the optimum value (in order of several thousands).

The data set used in (Y.L.Fablec and Y.Alliot, 1999) consists of aircraft trajectories with discrete time and represented with points sampled every 10 seconds. Some of the notation used in (Y.L.Fablec and Y.Alliot, 1999) includes:

- $z_0, z_1, \dots$ : the sequence of altitudes, where  $z_0$  corresponds to the altitude at time  $t_0 = 0$  sec. and  $z_i$  corresponds to the altitude at time  $t_i = 10 * i$  sec.
- Input the neural network:  $[z_i]$  - current altitude,  $[z_{i-n+1} - z_{i-n}, \dots, z_i - z_{i-1}]$  -  $n$  past vertical speeds.
- $[z_{i+1} - z_i]$  is speed to predict.

The authors also propose to use a sliding window in order to improve the algorithm and forecast positions in far future (when prediction can be changed



after flight path modifications). Author proposes to include the real points in the patterns in order to anticipate further positions with a slight delay. The method with sliding windows has been applied with a neural network with included 142 trajectories and 50 non-learned flight. Figures below, taken from (Y.L.Fablec and Y.Alliot, 1999), show examples of prediction.

Place figure 4 about prediction of the state-of-the-art here.

Results are not as good as expected, and show an oversmoothing of the trajectory. The reasons why neural networks do not seem to perform well on this specific problem are unclear as several parameters are interacting. A possible issue (which is also observed in the case of linear regression) is that the small number of samples used to forecast is not able to capture the variations of the trajectories shapes, ending with something that behaves much like a low-pass filter.

### *2.3. Imprecise Regression Modeling*

This section provides a discussion of the imprecise regression and regression on fuzzy data based on work of M. Serrurier and H. Prade (M.Serrurier and Prade, 2004). The authors' goal is to find a model as precise as possible that provides the most faithful description of the data. They suggest a precision function and define it for trapezoidal and triangular fuzzy sets and the description of the linear and non-linear imprecise functions. The authors also compare the imprecise regression with other types of fuzzy regression. According to (M.Serrurier and Prade, 2004), "the goal of the framework for imprecise regression is to overcome the learning biases by considering them as factors that have impact on the precision of the models rather than as a boundary to the effectiveness of the learning process".

Generally, the problem solved in (M.Serrurier and Prade, 2004) can be stated

as follows: A regression data consists of  $m$  pairs  $(\vec{x}_i, y_i)$ , where  $\vec{x}_i \in \mathcal{X}$  is a vector of  $n$  input variables and  $y_i \in \mathbb{R}$  is the real output variable. An imprecise function  $F$  associates the distribution on the possible values of the output to the inputs  $\vec{x}_i$ . Since (M.Serrurier and Prade, 2004) considers normalized possibility distributions and views them as fuzzy sets, they can denote the function  $F$  as  $F(\vec{x}_i) = \pi_i$ . In order to keep notation we give here the definitions as given in the original paper:

**Definition 1.** *Possibility distribution surface.*

Given  $\pi : \mathbb{R} \rightarrow [0, 1]$  a possibility distribution, the surface  $S(\pi)$  of this function is given by:

$$S(\pi) = \int_0^1 l(\pi_\alpha) d\alpha \quad (6)$$

with for  $\alpha > 0$ ,  $l(\pi_\alpha) = \mu(\{y \in \mathbb{R}, \pi(y) \geq \alpha\})$ . Here  $\mu$  is a Lebesgue measure and  $l(\pi_0) = \lim_{\alpha \rightarrow 0} l(\pi_\alpha)$

**Definition 2.** *Precision measure properties.*

Given  $\pi : \mathbb{R} \rightarrow [0, 1]$  a possibility distribution under  $\pi$  and  $Pr$  a precision function:

- $Pr : \mathbb{R} \rightarrow [0, 1]$
- $Pr(\pi) = 1$  if  $\exists! y \in \mathbb{R}$  such that  $\pi(y) = 1$  and  $\pi(y') = 0$  for  $y' \neq y$ .
- $Pr(\pi) = 0$  if  $\forall y \in \mathbb{R}, \pi(y) = 1$ .
- $Pr(\pi) \geq Pr(\pi')$  if  $S(\pi) \geq S(\pi')$ .

**Definition 3.** *Imprecise regression.*

Given a data set of  $m$  pairs  $(\vec{x}_i, y_i)$  and precision function  $Pr$ . The goal of imprecise regression is to find such  $F$  that maximizes the following:

$$R(F) = \frac{\sum_{i=1}^m \pi_i(y_i) * Pr(\pi_i)}{m} \quad (7)$$

where  $\pi_i = F(\vec{x}_i)$ .

Thus, authors consider two types of imprecise function: triangular sets and trapezoidal sets. More precisely, after applying to the sets the function have the following forms:

$$Pr(I_{l,m,r}) = \frac{\ln(1 + C * (r - l))}{C * (r - l)} \quad (8)$$

$$Pr(T_{a,b,c,d}) = \frac{\ln(1 + C * (b - a + d - c) + C(c - b))}{C * (b - a + d - c)} \quad (9)$$

To find the function  $F$  that maximize  $R(F)$  in Equation 7 they propose to use simulated annealing algorithm ((C.G.S.Kirkpatrick and M.P.Vecchi, 1983)). In order to use that technique they also define the neighborhood  $V$  of a function  $F$ . It is obtained by randomly adding and removing a fixed small value to all coefficients. It is due to the fact that simulated annealing is designed for discrete exploration of the space.

#### 2.4. Summary

Summarizing the section, we can say that all of the methods described in State-of-the-Art give correct results. At the end of the paper the reader will be able to see a comparison between these methods and our algorithm. Our new approach

of forecasting is based on functional regression and gives much better results than the multiple linear regression. This can be seen from the figures at the end of the paper as well as from the errors. The second method, based on neural networks shows a similar result to our work. In this paper we do not present the numeral comparison to that method, but we intend to do so in future work. The last method is based on solving the regression equation with imprecise modeling and shows bigger forecasting errors, which can be seen from the figure 18 at the end of the paper.

### **3. Functional Nature of Aircraft Trajectories**

#### *3.1. What are functional data?*

In mathematical and applied statistics, most of the analyses involve more than one observation for each observed object. In an increasing number of fields these observations are curves. Curves are often graphs of functions, since an observed intensity is available at each point on a line segment, a sector of a plane or portion of space. For this reason we assume that observed curves are functional data; statistical methods for analyzing such data are known under the name "functional data analysis" (FDA). One can find more information in Ramsay and Silverman's "Functional data analysis" (J.Ramsay and B.Silverman, 2005) including examples, methods and further references. Functional data analysis is also a general way of thinking, where the basic unit of information is the entire observed function rather than a string of numbers.

The main goal of functional data analysis is to study, most of the time, infinite - dimensional spaces of functions. Functional data analysis is related to statistics theory and is a branch of it. Functional regression, Principal component analysis

(PCA), canonical analysis, least square problems lie in the heart of FDA. FDA has to cope with the following issues:

- formulation of the problem in a way amenable to statistical thinking and analysis;
- definition and representation of qualitative and quantitative characteristics of the data;
- expansion and decomposition of the data into basis functions;
- development of criteria and evaluation of the effectiveness of statistical methods;
- construction of models for observed data that capture relations between observation variables.

In general, we consider a collection of functions, rather than just a single function. Since we cannot get continuous observations of aircraft trajectories, we assume that trajectories are sampled approximations of the real functions. We illustrate that fact in the next section. The observations  $f$  of the real function consist of  $m$  pairs of recordings, such as  $f = \{f(t_j)\}_{j=1}^m$ . Here,  $\{t_j\}_{j=1}^m$  is a sample of  $[a, b]$  which is a time interval over which  $f$  is observed. The samples  $t_j$ ,  $1 \leq j \leq m$  do not need to be evenly spaced. Considering the process of altitude sampling, the corresponding data is assumed to originate from the process:

$$Alt(t) = f(t) + \epsilon(t) \tag{10}$$

where  $f(t)$  is deterministic function and  $\epsilon$  is an error with zero mean.

Figure 5 shows an example of the type of data that will be considered in simulation studies below. Here the data consist of the observation of 20 aircraft trajectories altitude during take-off in Toulouse airport. The samples are not evenly spaced over time, and they were obtained from ATC radar with time increments from 3 to 30 seconds. The data consist of the sample of 20 functional observations of aircraft altitude  $Alt_i(t)$  during take-off.

Place the figure 5 about example of altitude of 20 aircraft trajectories during take-off here.

The standard mathematical statistics applied to functional data (J.Ramsay and B.Silverman, 2005) is similar to the classical summary of statistics for discrete data. The mean function defined by:

$$\bar{x}(t) = \frac{1}{N} \sum_{k=1}^N x_k(t), \quad (11)$$

and it is the average of the functions point-wise across duplications. Similarly the variance is:

$$var_x(t) = (N - 1)^{-1} \sum_{k=1}^N (x_k(t) - \bar{x}(t))^2, \quad (12)$$

and the standard deviation is the square root of the variance function.

Figures 6 and 7 show complete 50 trajectories altitude and the mean of them. Let us mention that the mean curve is not a real trajectory. Each value of the mean corresponds to the mean of the observations of the trajectories at a given time.

Place figures 6 and 7 aircraft trajectories and their mean here.

The covariance function summarizes the dependence of records across different argument values and is computed as:

$$cov_x(t_1, t_2) = (N - 1)^{-1} \sum_{k=1}^N (x_k(t_1) - \bar{x}(t_1))(x_k(t_2) - \bar{x}(t_2)) \quad (13)$$

The associated correlation function is given by:

$$corr_x(t_1, t_2) = \frac{cov_x(t_1, t_2)}{\sqrt{var_x(t_1)var_x(t_2)}} \quad (14)$$

These are functional analogues to the variance, covariance and correlation matrices, respectively in multivariate data analysis.

### 3.2. Aircraft Trajectories as Functional Data

In this section we propose a new approach to describe aircraft trajectories determined by the flight plan, but it also depends on many other factors, such as weather conditions, behavior of pilots and dispatchers instructions, the physical properties of the aircraft ; we will consider these factors as random variables. This gives us a reason to is why we will consider the trajectory as a random vector-function of time. Moreover, the vector-function is at least twice differentiable and has bounded energy because of the physical properties and the presence of velocity and acceleration. In particular, a model of observations for the set of aircraft altitudes, described in the previous section, is based on samples as follows:

$$Alt(t_j) = f(t_j), j = 1, \dots, m \quad (15)$$

where  $f(t_j)$  is a radar sample of altitude at time  $t_j$ . Let  $t_0$  be a point in the future. We would like to forecast the position of the aircraft at  $t_0$ , or in other words, we want to evaluate the value of the regression function at time  $t_0$ . Denote this value as  $f^*(t_0)$ . Predicted value  $f^*(t_0)$  has to be sufficiently close to the real value  $Alt(t_0)$  to be of any practical value. The response of the model at any

point of time is a random variable, because of the randomness of  $\epsilon$  and therefore the randomness of the sample. Moreover, different samples will lead to different models and therefore to a different response at  $t_0$ .

In a probabilistic sense, saying that the estimator  $f^*(t_0)$  is close to  $Alt(t_0)$  means that the Mean Square Error (MSE) is small:

$$MSE = E|f^*(t_0) - Alt(t_0)|^2. \quad (16)$$

In terms of Bias/Variance, the MSE can be shown as in (G.Lebanon, 2010):

$$MSE = ||\text{Bias}(f^*(t_0))||^2 + \text{trace}(\text{Variance}(f^*(t_0))) \quad (17)$$

Let us now give an intuitive description of *Bias/Variance dilemma*. The *Bias* parameter measures the deviation of predicted value  $f^*(t_0)$  of the model to the real value of function  $Alt(t_0)$ . The *Variance* measures the sensitivity of predicted value to the sample. The *Bias/Variance dilemma* is very important problem in data modeling. Another one issue is model complexity. High model has a low *Bias*, but a large *Variance*, while low complexity model has a high *Bias*, but a low *Variance*. As a consequence of it, there exists an optimal number of degree of freedoms, which has to be used for working on the best possible model.

For theoretical description of estimation of error via *Bias/Variance dilemma* we address to *R.Doursat et al* ((S.R.Doursat and E.Bienenstock, 1992)):

$$E[f^*(t) - E[Alt|f(t)]]^2 = \text{Bias} + \text{Variance}^2 \quad (18)$$

where  $\text{Bias} = (E[f^*(t) - E[Alt|f(t)]]^2$  and  $\text{Variance} = E[f^*(t) - E[f(t)]]$ . As we said above, an unbiased estimator can have a high MSE if variance is large and  $f^*(t)$  can be highly sensitive to the data and be far from the regression.



One of the methods of model parameters selection is cross-validation procedure, particularly  $k$ -fold cross-validation and leave-one-out cross-validation.

Place figure 8 about leave-one-out cross validation procedure here.

Cross-validation procedure is a technique for generalization assessing of statistical models to an independent data set. Mainly it uses prediction algorithms to estimate the prediction accuracy of the model.  $k$ -fold cross-validation is a technique which takes a set of  $m$  examples and then partitions them into  $K$  sets (or folds) of size  $m/K$ . Then, each fold is retained as the validation data for testing the model and the remaining  $k - 1$  folds are then used as training data. The cross-validation process is then repeated  $k$  times. The advantage of  $k$ -fold cross-validation is that all the examples from the data set are eventually used for both training and testing. Particular case of  $k$ -fold cross-validation is leave-one-out cross-validation, where  $k = 1$  (figure 8).

Another important point is the following one: the data are long term discrete observations coming from ATC radars. We need preprocessing the data by doing interpolation, in order all trajectories are sampled at same times. Moreover, the data we have consists of  $3D$  space coordinates, it means that we are dealing with multivariate interpolation, which is interpolation of function of more than one variable. So, the kernel idea of assuming aircraft trajectories as functions can be summarized on figure 9.

Place figure 9 about the benefits of the algorithm here.

All previous aspects turn us to consider the aircraft trajectories as functions, moreover as a set of random functions of a random process, which is the topic of the next chapter.

### 3.3. Aircraft Trajectories as a sample of Hilbert Stochastic Process

Let us begin from the definition of random process. A lot of literature has been written about stochastic processes, we recommend (I.Gikhman and A.Skorokhod, 1996), (J.Doob, 1990), (A.Papoulis and S.Pillai, 2001).

**Definition 4.** *Given a probability space  $(\Omega, \mathfrak{F}, \mathbb{P})$ . A function of two variables  $X(t, \omega), t \in T, \omega \in \Omega$  taking values in a metrical space  $\mathfrak{X}$ ,  $\mathfrak{F}$  measurable as a mapping of  $\omega$  for each  $t \in T$  is called a random process.*

The definition of a Hilbert random process can be given as in (I.Gikhman and A.Skorokhod, 1996).

**Definition 5.** *Let  $(\Omega, \mathfrak{F}, \mathbb{P})$  be a probability space. The Hilbert space  $\mathfrak{L}_2 = \mathfrak{L}_2\{\Omega, \mathfrak{F}, \mathbb{P}\}$  is the set of complex random variables  $\zeta = f(\omega), \omega \in \Omega, f \in \mathfrak{F}$ , for which  $E|\zeta|^2 < \infty$ .*

*Inner product in  $\mathfrak{L}_2$  is defined as following:*

$$\langle \zeta, \eta \rangle = E\{\zeta \bar{\eta}\}, \quad \zeta, \eta \in \mathfrak{L}_2 \quad (19)$$

*Corresponding norm  $\|\zeta\|$  of the random variable  $\zeta$  is defined by:*

$$\|\zeta\| = \{E|\zeta|^2\}^{1/2} = \left\{ \int_{\Omega} |\zeta|^2 d\mathbb{P} \right\}^{1/2} \quad (20)$$

Two random variables  $\zeta$  and  $\eta$  are orthogonal, iff

$$\langle \zeta, \eta \rangle = 0 \quad (21)$$

For the real random variable  $\zeta$ , the square of the norm  $\|\zeta\|^2$  coincides with the second order moment, i.e.  $\|\zeta\|^2 = E\zeta^2$ , and if  $E\zeta = 0$  it coincides with variation.

**Definition 6.** Let  $\zeta(t, \omega)$ ,  $t \in T$  be a random process (of variable  $\omega$ ), and let us denote  $\zeta_t$  the function defined for any  $\omega$  by  $\zeta_t(\omega) = \zeta(t, \omega)$ .  $\zeta$  is said to be a random Hilbert process iff  $\zeta_t \in \mathfrak{L}_2$  and, for any  $t \in T$ , :

$$E|\zeta_t|^2 < \infty \quad (22)$$

i.e. a random Hilbert function can be regarded as a Hilbert space-valued function depending on  $t \in T$ .

In case of aircraft trajectories  $\Omega$  plays a role of a set of flight identification numbers (flight ID). Let us define an aircraft trajectory as a mapping from time interval  $[a, b]$  to the space  $\mathbb{R}^3$ . Assume,  $f(t, \omega_0) = f(t) = f(x(t), y(t), z(t))$ ,  $t \in [a, b]$  is aircraft trajectory function with flight ID corresponding to  $\omega_0$  and where  $x(t)$ ,  $y(t)$  and  $z(t)$  are longitude, latitude and altitude functions respectively. Let us remind the reader that, as mentioned above, the aircraft trajectories are at least twice differentiable and have bounded energy. et us now consider a set of trajectories over a fixed time interval  $[a, b]$  as a set of random functions belonging to 3 dimensional Hilbert stochastic process.  $f$  should satisfy conditions of Hilbert stochastic process and has finite energy  $(\int_a^b \|f(t)\|^2 dt)^{1/2} < \infty$ . Finally, at time  $t \in [a, b]$  there exists a random variable  $\zeta(\chi)$ , which is a set of coordinates of trajectories at time  $t$  and  $E(\|\zeta(t)\|)^2 < \infty$ .

Now, we also can define the trajectory by adding the  $\ll$ -dimensional speed. In this case, the trajectory is a mapping from  $[a, b]$  to  $\mathbb{R}^6$ . So, we will focus on trajectories that belong to the Sobolev space of order 1 of square integrable mappings with square integrable derivatives.

### 3.4. Wavelets

Let us now give a brief introduction to the wavelets. We can construct a basis for all functions on  $(-\infty, \infty)$  that are square-integrable by choosing a suitable mother wavelet function  $\phi$  and then considering all dilations and translation of the form (see details for example in (J.Ramsay and B.Silverman, 2005), (Daubechies, 1992), (Daubechies, 1990))

$$\phi_{jk}(t) = 2^{j/2} \phi(2^j t - k) \quad (23)$$

for integers  $j$  and  $k$ . We construct the mother wavelet to ensure that the basis is orthogonal in  $\mathcal{L}_2$ . Typically, the mother wavelet has compact support, and hence so does all the basis functions. The wavelet basis idea is easily adapted to deal with functions defined on a bounded interval, most simply if periodic boundary conditions are imposed. The wavelet expansions of a function  $f$  gives a multiresolution analysis in the sense that the coefficients of  $\phi_{jk}$  yield information about  $f$  near position  $2^{-j}k$  on scale  $2^{-j}$ , i.e. at frequencies near  $c2^j$  for some constant  $c$ . Thus, wavelets provide a systematic sequence of degrees of locality.

In contrast to Fourier series, wavelet expansions cope well with discontinuous or rapid changes in behavior; only those basis functions whose support includes the region of discontinuity or other bad behavior are affected. This property, as well as a number of more technical mathematical results, means that it is often reasonable to assume that an observed function is well approximated by an economical wavelet expansion with few nonzero coefficients, even if it displays sharp local features. Suppose a function  $x$  is observed without error at  $n = 2^M$  regularly spaced points on interval  $T$ . Just as with the Fourier transformation, there is a discrete wavelet transform (DWT) which provides  $n$  coefficients closely re-

lated to the wavelet coefficients of the function  $x$ . We can calculate the DWT and its inverse in  $O(n)$  operations, even faster than the  $O(n \log n)$  of the Fast Fourier Transform (FFT). As a consequence, most estimators based on wavelets can be computed extremely quickly, many of them in  $O(n)$  operations.

Place figures 10, 11 and 12 about aircraft coordinates and their wavelet transforms here.

Figures 10, 11 and 12 show an example of aircraft trajectory's  $x$ ,  $y$  and  $z$  coordinates and the scalogram of their continuous wavelet transform (CWT) using daubechies 4 wavelet. As it seen from these figures, there are picks of wavelet decomposition at points when an aircraft makes turns (maneuvers). It means that wavelets feel the major changes in directions of speed-vector of trajectory function (singularity or discontinuity points). Thus, applying the wavelet decomposition in our work helps us to avoid useless information in construction of the shape of the trajectory. What is more, we can use as many points in our regression model as we need in order to minimize the MSE and control the generalization behavior of the model.

*Wavelets in Sobolev space.* Since we are using the speed of aircraft in our modeling, we need to define the norm of the wavelets contributed with the first derivative of them or in other words define the norm in Sobolev space. Sobolev space is a functional space consisting of functions from Lebesgue spaces  $L_p$ , having generalized derivatives of a given order in  $L_p$ . With  $1 \leq p \leq \infty$  Sobolev spaces are Banach spaces, and when  $p = 2$ , the Sobolev spaces are Hilbert spaces. We recommend following literature about wavelets in Sobolev space (R.Jia (2003),

**Definition 7.** Let  $s \in \mathbb{N}$ . The function  $f \in L_2(\mathbb{R})$  belongs to the Sobolev space  $W^s(\mathbb{R})$ , if it is  $s$ -times weakly differentiable, and if  $f^{(j)} \in L_2(\mathbb{R})$ ,  $j = 1, 2, \dots, s$ .

In a Sobolev space the norm is given by :

$$\|f\|_{W^s(\mathbb{R})}^2 = \|f\|_{L_2(\mathbb{R})}^2 + \|f^{(s)}\|_{L_2(\mathbb{R})}^2. \quad (24)$$

As shown in the previous section, any function  $f \in L_2(\mathbb{R})$  can be represented as a series (convergent in  $L_2(\mathbb{R})$ ):

$$f(\cdot) = \sum_{k \in \mathbb{Z}} c_k \phi_{0,k}(\cdot) + \sum_{j=0}^{\infty} \sum_{k \in \mathbb{Z}} c_{jk} \psi_{jk}(\cdot) \quad (25)$$

where  $c_k, c_{jk}$  are wavelet expansion coefficients, and

$$\|f\|_{L_2(\mathbb{R})}^2 = \sum_{k \in \mathbb{Z}} c_k^2 + \sum_{j=0}^{\infty} \sum_{k \in \mathbb{Z}} c_{jk}^2. \quad (26)$$

As shown in (R.Jia, 2003), (V.Meyer, 1992), a function  $f$  lies in  $W^s(\mathbb{R})$  if and only if

$$\sum_{k \in \mathbb{Z}} c_k^2 + \sum_{j=0}^{\infty} \sum_{k \in \mathbb{Z}} 2^{sj} c_{jk}^2 < +\infty. \quad (27)$$

Moreover, the discrete equivalent norm of wavelets decomposition in Sobolev space  $W^s(\mathbb{R})$  is:

$$\|f\|_{W^s(\mathbb{R})}^2 \approx \sum_{k \in \mathbb{Z}} c_k^2 + \sum_{j=0}^{\infty} \sum_{k \in \mathbb{Z}} 2^{sj} c_{jk}^2, \quad (28)$$

where  $s$  is the smoothness order of the Sobolev space.

#### 4. Trajectory Smoothing and Compressing using Wavelets.

As explained in previous chapter, the problem of bias/variance dilemma is a critical issue in solving the linear regression. Another issue, arising when per-

forming regression based prediction is the high level of correlation between successive samples in a trajectory. Since most of the time aircraft are following piecewise straight line paths, nearly all samples are linearly dependent, thus add no additional information to the model; in the same time, the dimension of the linear system to be solved for regression estimation increases, which induces a corresponding increase in computational effort and decrease in numerical accuracy. In both cases, a clever reduction of redundancy will have a positive impact on the outcome of the prediction. It exists an optimal way of reaching this goal: the principal component expansion of Karhunen-Loeve. Given a set of  $P$  sampled trajectories of size  $N$ ,  $(Y_{i1}, \dots, y_{iN})_{i=1, \dots, P}$ , principal components extraction starts with the empirical correlation matrix:

$$\Sigma = (\Sigma_{ij} = P^{-1} \sum_{k=1}^P Y_{ki} Y_{kj})_{i,j=1, \dots, N} \quad (29)$$

where the trajectories are assumed to have zero mean.

The principal components are then obtained by finding the eigenvectors of  $\Sigma$ , ordered by decreasing eigenvalues. It can be shown that truncating the principal components expansion to a given order  $Q \leq N$  yields an optimal representation in the sense that no other expansion of the same length will have a lower residual variance. Furthermore, apart from the mean that plays a special role, the principal components represent salient features of the trajectory set.

Unfortunately, the process of finding eigenvectors and eigenvalues of a matrix has complexity  $O(n^3)$  and is not really usable for long trajectories. Wavelet analysis gives a result close to optimal representation for a complexity of only  $O(n \log n)$  which is much more tractable. Actually, most of modern image compression and denoising algorithms are based on wavelets. Choosing an optimal

number of wavelet coefficients for the regression is a critical task to solve the problem. It is possible to find such optimal number of wavelet coefficients, which efficiently characterize trajectories curves, using different statistical methods like Akaike information criterion, Bayesian information criterion or K-fold cross validation procedure. In the work presented here, the optimal number of coefficients had been chosen intuitively.

Place figure 13 about smoothing and compressing properties of wavelets here.

Figure 13 shows the effect of using a different number of wavelet coefficients. The curve (regular line) of the original signal consists of 128 points. These data are not experimental data, but artificial ones, made in order to explain the reader the effect of using different number of wavelet in the decomposition of the function. This curve contains sharp turns and randomness, which is the case of aircraft trajectories. The dashdot line ('-.') shows the curve of the inverse wavelet decomposition with only considering first 16 coefficients of discrete wavelet transform (daubechies 4 wavelet) of the original signal. It is clearly seen, that considering such number of coefficients extracts the tendency of the signal, which is relevant in prediction problems. The dashed line ('-') demonstrates the result of reconstruction of the signal, only considering first 32 wavelet coefficients (daubechies 4 wavelet). In this example, signal reconstruction with 32 coefficients smoothes the curve and removes sharp variations.

Now, having defined all the basic concepts and definitions related to our working field and explained the reasons of our approach with its benefits, we switch on the constructing and solving the main prediction algorithm.



## 5. Solving Functional Linear Regression Using Wavelets

### 5.1. A New Approach to Trajectory Prediction

We present here a new approach to the short to mid-term aircraft trajectory prediction problem. It is based on modeling past trajectories, and doing local linear functional regression for finding a kind of probable standard trajectories for the future of the present one, and solving so the short to mid-term Trajectory Prediction (TP) problem ((C.Gong and D.McNally, 2004)). By definition, an aircraft trajectory is a mapping from a time interval  $[a, b]$  to the space  $\mathbb{R}^3$  (sometimes, it is convenient to add speed, so that the resulting extended state space is  $\mathbb{R}^6$ ). Such a trajectory is the observed result of a complex evolution process that involves flight dynamics, external actions (pilot, ATC) and atmospheric factors (wind, temperature ...). The complete description of the trajectory, using all these factors is generally not possible, because many influencing factors are unknown (aircraft mass, local wind, etc ...), so a less accurate but tractable model has to be chosen. For the purpose of short term prediction, a linear controlled model :

$$\mathbf{X}'(t) = \mathbf{A}X(t) + u(t) \quad (30)$$

is accurate enough. In this model,  $X$  is the state vector,  $\mathbf{A}$  is the matrix describing the evolution of the system and  $u$  is the command. This last function  $u$  is modeled as a point process, which is satisfied by real traffic. Based on this observation, we will focus on trajectories that belong to the Sobolev space of square integrable mappings with square integrable derivatives (in the sense of distributions) up to order 3.

## 5.2. *Strengths and Weaknesses*

*Strengths.* The algorithm we propose uses the combination of the latest advances in applied mathematics and well developed techniques. It requires only past observed data to build an efficient prediction model. One of the main advantages of the method is localization of the data. It allows extracting the closest trajectories with the similar characteristics and similar behavior. A longer time period data will be the positive side of the algorithm, despite the different weather conditions. As mentioned before, localization of the data allows extracting trajectories with similar characteristics, which collaterally contains the similar weather conditions. Moreover, later in this paper the reader can see that we assign the weights to each trajectory in our model, which affects to pay more attention on closest samples. Next critical task is to choose an appropriate space of functions. Such space of functions should contain the basis with a fast decomposition in couple with fast inverse transform. Using wavelet decomposition allows to extract all necessary information about the shape of trajectory in a  $O(n \log n)$  operations. Moreover, wavelet coefficients decrease fast, which is a benefit when computing a large amount of data.

*Weaknesses.* Certainly, like many other prediction methods, our approach has its weaknesses. An important constraint is the need of past, not outdated data. The method does not take into account important physical, aeronautical, or even weather parameters of the various handled aircraft.

Actually, two different prediction approaches exist. First one is based on flight dynamics equations and contains physical aspects of an aircraft. Such models use mass of an aircraft; it's centering, fuel consumption, etc. The lack of such kind of approach is realistic impossibility of delivering the exact values of physical

parameters presently. The second approach consists of regression based methods; some of them described in "State-of-the-Art". No doubt, the main limitation of the regression based algorithms is the fact, that they require enough amounts of data, or statistics, to produce an efficient result. It must be remarked that with only several days of traffic or on the routes with very low frequencies, regression based algorithms fail with high probability. It can be explained by insufficiency of the data for the model learning. One other issue of our approach is not taking into account the weather conditions as a variable. But, for solving that problem it is necessary to obtain weather observations over longer period, at least one year. It is not obvious to operate with such amount of complicated data and to integrate a new variable to the prediction model. Besides, the present algorithm should be optimized from the computational point of view. Anyway, in the future work seasonal weather conditions will be taking into account and the model will be corrected using weather information.

### 5.3. *Functional Linear Model for Functional Response*

First, let us give a brief summary to the functional linear models with functional response. This part considers an estimation procedure for a functional linear regression model where both predictor (realizations of a functional predictor) and response (functional response) variables are functions (A.Aguilera and M.Valderrama, 2005). This model has been studied by Yao, Muller and Wang (B.F. Yao, 2005), (F.Ferraty and P.Vieu, 2006), who proposed an estimation approach for sparse and irregular longitudinal data based on a nonparametric estimation for the eigenfunctions of the sample covariance operators associated to both predictor and response functional variables.

Let us consider a functional predictor variable  $\{X_\omega(t), t \in T, \omega \in \Omega\}$  and

supposed to be a functional response  $\{Y_\omega(s), s \in S, \omega \in \Omega\}$ , where  $\{\Omega, \mathfrak{F}, \mathbb{P}\}$  is a probability space,  $T$  and  $S$  are compact time intervals in  $\mathbb{R}$ ,  $X_\omega(t)$  and  $Y_\omega(s)$  are Hilbert random processes. Samples consist of pairs of random trajectories  $\{x_\omega(t), y_\omega(s), \omega = 1, \dots, N\}$ , that are realizations of the functional predictor and functional response respectively.

The functional linear regression model used to estimate functional response  $Y(s)$  from the functional predictor  $X(t)$  is:

$$\hat{Y}_\omega(s) = \alpha(s) + \int_T X_\omega(t) \beta(s, t) dt + \epsilon_\omega(s), \quad s \in S, \quad (31)$$

where  $\{\epsilon_\omega, \omega \in \Omega\}$  are independent and identically distributed random errors with zero mean and  $E\{\epsilon_\omega^2(s)\} = \sigma_Y^2$ ,  $\beta(t, s)$  is a square integrable matrix valued kernel. Without loss of generality, suppose that predictor and response variables are centered ( $\mu(\hat{X}(t)) = \mu(\hat{Y}(s)) = 0$ ) by subtracting their sample average functions  $\bar{x}_\omega(t) = x_\omega(t) - \mu_X(t)$  and  $\bar{y}_\omega(s) = y_\omega(s) - \mu_Y(s)$ . Then, the problem is reduced to a well-known approximation which is formalized as estimation of the conditional mean function:

$$E\{Y(s)|x_\omega\} = \int_T \beta(t, s) x_\omega(t) dt + \epsilon_\omega(s). \quad (32)$$

So, the prediction (i.e. the determination of the response  $Y(s)$ ) now only depends on the arbitrary function  $\beta(t, s)$ . Now, the goal is to estimate the parameter kernel  $\beta(t, s)$ . This is an ill-posed problem due to the infinite dimension of predictor and response realizations.

#### 5.4. Estimation of the Regression Function

The first estimation of parameter function  $\beta(t, s)$  can be obtained by assuming that both predictor and response sample curves belong to a finite dimension space generated by two different basis  $\{\phi_p, p = 1, \dots, P\}$  and  $\{\psi_q, q = 1, \dots, Q\}$ . Such sample trajectories can be expanded as:

$$x_\omega(t) = \sum_{p=1}^P a_{\omega p} \phi_p(t), \quad y_\omega(s) = \sum_{q=1}^Q b_{\omega q} \psi_q(s) \quad (33)$$

where  $\{a_{\omega p}, p \in P\}$ ,  $\{b_{\omega q}, q \in Q\}$  are coefficients of decomposition. The expression of the parameter kernel function considers as the double expansion as follows:

$$\beta(t, s) = \sum_{p=1}^P \sum_{q=1}^Q \beta_{pq} \phi_p(t) \psi_q(s) \quad (34)$$

Then, the model in( 32) is equivalent to the following multivariate linear regression model:

$$b_{\omega q} = \sum_{p=1}^P b_{pq} \sum_{r=1}^P a_{\omega r} \phi_{pr} + \epsilon_{\omega q}, \quad q = 1, \dots, Q \quad (35)$$

or in matrix form  $B = A\Psi\beta + \Upsilon$ .

Then, the estimation of the parameter function can be obtained from the least mean squares estimation of its matrix of coefficients  $\hat{\beta} = ((A\Psi)'(A\Psi))^{-1}(A\Psi)'B$ :

$$\hat{\beta}(t, s) = \sum_{p=1}^P \sum_{q=1}^Q \hat{\beta}_{pq} \phi_p(t) \psi_q(s). \quad (36)$$

### 5.5. Linear Regression Based on Weighted Distance-based Regression

This section considers the problem of non-parametrically predicting a functional response variable  $Y$  from a functional predictor  $X$ . Let  $(X, Y)$  be a random element defined on compact time intervals  $T = [a, b]$  and  $S = [c, d]$  respectively, where  $X$  and  $Y$  are random variables of a functional space. Consider the problem of predicting the functional response variable  $Y$  from the functional predictor  $X$ . Assume that  $n$  independent and identically distributed observations  $(x_i, y_i)$ ,  $i = 1, \dots, n$  are given from  $(X, Y)$  as a training set. The linear regression model, defined as in (A.Aguilera and M.Valderrama, 2005), is:

$$m(X) = E\{Y|X\} = \alpha(t) + \int_T X(t)\beta(t, s)dt, \quad y_i = m(X) + \epsilon_i, \quad (37)$$

where the  $\epsilon_i$  are i.i.d. errors with zero mean. Parameter function  $\beta(t, s)$  is a square integrable matrix valued mapping and  $\alpha(t)$  is the square integrable mapping. Ferraty and Vieu ((F.Ferraty and P.Vieu, 2006)) consider this linear regression model as parametric because we only need a finite number of functional elements to describe it. They also address nonparametric functional regression model where only a few regularity assumptions are made on the regression function  $m(x)$ . For a given Kernel  $K$ , they propose the following estimator of  $m(x)$ :

$$\hat{m}_{K(x)} = \frac{\sum_{i=1}^n K(d(x, x_i)/h) y_i}{\sum_{i=1}^n K(d(x, x_i)/h)} = \sum_{i=1}^n \omega_i(x) y_i, \quad (38)$$

where  $\omega_i(x) = K(d(x, x_i)/h) / \sum_{j=1}^n K(d(x, x_j)/h)$ ,  $K$  is a kernel function with compact support, the bandwidth  $h$  is the smoothing parameter and  $d(x, x_i)$  is a semi-metric in the functional space  $\mathbb{F}$  of the data  $x_i$ . Authors prove that  $\hat{m}(x)$  is a consistent estimator (in the sense of almost complete convergence) of  $m(x)$

under regularity conditions on  $m$ ,  $X$ ,  $Y$  and  $K$ . In "Nonparametric functional data analysis" Ferraty and Vieu list several interesting open problems concerning nonparametric functional regression. The first answer to such regression problem is given by Baillo and Grane((A.Baillo and A.Grane, 2007)) as Boj, Delicado and Fortiana mentioned. They propose a natural extension of the finite dimensional local linear regression by solving the following problem:

$$\min_{\alpha, \beta} \sum_{i=1}^n \omega_i(x) \left( y_i - \alpha(t) - \int_T (x_i(t) - x(t)) \beta(t, s) dt \right)^2, \quad (39)$$

where local weights  $\omega_i(x) = K(\|x - x_i\|/h) / \sum_{j=1}^n K(\|x - x_j\|/h)$  are defined by means of  $L_2$  distances (assumed that all functions are from  $L_2$ ). The problem that has to be addressed now is to find the right basis for representation. The magnitude of the coefficients of the expansion has to decrease fast in order to keep truncation efficient. Wavelets have proven to have such a property for a wide class of functions.

#### 5.6. *The heart of the method: solving Local Linear Weighted-Distance based Regression Using Wavelets in Sobolev Space*

Let us remind the reader that aircraft trajectories belong to some Hilbert stochastic process  $H$ , defined on time interval  $T$ . We assume that target trajectory to predict belongs to  $H$  with  $\omega_0$ , in other words it is a function  $f_0 \in L_2(\mathbb{R}^3)$ , such that for any  $t \in [a, b]$   $f_0 = f(\omega_0, t)$  for some  $f \in H$ . To predict future position of the trajectory  $f_0$  on interval  $[c, d]$ , where  $c \geq b$ , we use past known observations of the trajectories coming from  $H$ , i.e. it should be the same class of trajectories going the same direction (Origin-Destination) in the past. All these observations are samples of stochastic process  $H$ . Here and later we assume that all trajectories

are synchronized by the time. Then, we need to divide all the trajectories into two parts, predictor and response. Predictor consists of trajectories, defined on  $[a, b]$  time interval and response on  $[c, d]$ , respectively. For midterm forecasting, the forecasting time interval  $[c, d]$  varies from 20 to 30 minutes. We define the general functional linear regression model as follows:

$$\hat{Y}(s) = \alpha(s) + \int_a^b \beta(s, t) X(t) dt + \epsilon(\omega) \quad (40)$$

where  $\alpha(s)$  is a square integrable mapping,  $\beta(s, t)$  is a kernel function which is a square integrable matrix-valued mapping defined on  $[a, b] \times [c, d]$ .  $X_i(t)$  is a trajectory from predictor part and  $\epsilon_i$  is an error. The problem is to find the optimal couple  $(\hat{f}(t), \hat{\beta}(t, s))$  minimizing the mean square error between  $Y$  and  $\hat{Y}$ .

So,  $X(t)$  is a part of  $H$  on  $[a, b]$  and  $Y(s)$  is a part of  $H$  on  $[c, d]$ . In previous section without loss of generality it has been assumed that  $H$  is centralized, i.e.  $\alpha(s) := 0$  and  $E|\epsilon| = 0$ . We start by solving the regression by expanding both  $X$  and  $Y$  on wavelet basis:

$$X_i(t) = X(\omega_i, t) = \sum_{j \in \mathbb{Z}} a_j \psi_j^x(t), \quad t \in [a, b] \quad (41)$$

and

$$Y_i(s) = Y(\omega_i, s) = \sum_{k \in \mathbb{Z}} b_k \psi_k^y(s), \quad s \in [c, d], \quad (42)$$

where  $\{a_j\}_{j \in \mathbb{Z}}$  and  $\{b_k\}_{k \in \mathbb{Z}}$  are wavelet coefficients corresponding to  $\Psi^x = \{\psi_j^x\}_{j \in \mathbb{Z}}$  and  $\Psi^y = \{\psi_k^y\}_{k \in \mathbb{Z}}$  basis respectively. Different wavelet basis were used in numerical computations. There are: Haar wavelet, Daubechies 4 and Coiflet 6. Since



we are working in a Sobolev space, we also need to compute the norm of wavelets considering the first derivatives. Here we again address to (R.Jia, 2003), (V.Meyer, 1992):

**Definition 8.** *Function  $f$  lies in  $W^s(\mathbb{R})$  if and only if*

$$\sum_{k \in \mathbb{Z}} c_k^2 + \sum_{j=0}^{\infty} \sum_{k \in \mathbb{Z}} 2^{sj} c_{jk}^2 < +\infty.$$

*Moreover, the discrete equivalent norm of wavelets in Sobolev space  $W^s(\mathbb{R})$  is given by:*

$$\|f\|_{W^s(\mathbb{R})}^2 \approx \sum_{k \in \mathbb{Z}} c_k^2 + \sum_{j=0}^{\infty} \sum_{k \in \mathbb{Z}} 2^{sj} c_{jk}^2$$

*where  $s$  is the smoothness order of the Sobolev space.*

Kernel function  $\beta(t, s)$  can be expanded as a double sum as following:

$$\beta(t, s) = \sum_{j \in \mathbb{Z}} \sum_{k \in \mathbb{Z}} \kappa_{jk} \psi_j^x(t) \psi_k^y(s). \quad (43)$$

For each trajectory in regression we assign a weight  $\lambda_i = \Lambda(d(f_0; X_i))$ , where  $d$  is a semi-metric window function. Such window functions take into account the distance between the trajectories from the learning data set to predicted one. Using a window function we set the weights to the trajectories from training set that have a similar behavior as the "real" one. An example of such a function used in this work is the Epanechnikov kernel:

$$\Lambda(t) = \frac{3}{4}(1 - t^2)I_{[-1;1]}(t). \quad (44)$$

Another choice is Gaussian kernel:

$$\Lambda(t) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{t^2}{2}\right). \quad (45)$$

Such an approach improves the efficiency of the trajectory selection from the data set by the mean of the window function value.

Using basis orthonormality property, the regression problem becomes now to find the minimum of the following sum, where the expansions have been truncated to a fixed rank:

$$\begin{aligned} & \min_{\beta(t,s)} \left\{ \sum_{i=1}^N \lambda_i \left\| Y_i(t) - \int \beta(t,s) X_i(s) ds \right\|^2 \right\} = \\ & \min_{k_{jk}} \left\{ \sum_{i=1}^N \lambda_i \left\| \sum_{j=1}^P b_j^i \psi_j(t) - \int_{\tau_X} \sum_{j=1}^P \sum_{k=1}^Q k_{jk} \phi_k(s) \psi_j(t) \sum_{k=1}^Q a_k^i \phi_k(s) ds \right\|^2 \right\} = \\ & \min_{k_{jk}} \left\{ \sum_{i=1}^N \lambda_i \left\| \sum_{j=1}^P b_j^i \psi_j(t) - \sum_{j=1}^P \sum_{k=1}^Q k_{jk} a_j^i \psi_j(t) \int_{\tau_X} \phi_j(s) \phi_k(s) ds \right\|^2 \right\} = \\ & \min_{k_{jk}} \left\{ \sum_{i=1}^N \lambda_i \left\| \sum_{j=1}^P b_j^i \psi_j(t) - \sum_{j=1}^P \sum_{k=1}^Q k_{jk} a_j^i \psi_j(t) \right\|^2 \right\} = \\ & \min_{k_{jk}} \left\{ \sum_{i=1}^N \lambda_i \left\| \sum_{j=1}^P [b_j^i - \sum_{k=1}^Q k_{jk} a_j^i] \psi_j(t) \right\|^2 \right\} = \\ & \min_{k_{jk}} \left\{ \sum_{i=1}^N \lambda_i \sum_{j=1}^P (b_{ji} - \sum_{k=1}^Q a_{ki} k_{jk})^2 \right\} \end{aligned}$$

This expression was solved using Singular value Decomposition (SVD) (J.Demmel, 1997).

Then,  $\hat{\alpha}(s)$  can be expressed by the following formula:

$$\hat{\alpha}(s) = \bar{Y}(t) - \int_{\tau_X} \hat{\beta}(t, s) \bar{X}(s) ds = \sum_{j=1}^P [\bar{b}_j - \sum_{k=1}^Q \hat{k}_{jk} \bar{a}_k] \psi_j(t). \quad (46)$$

Where  $\bar{b}_k$ ,  $\bar{a}_j$  are means of wavelet coefficients,  $\hat{\beta}$  is the result of minimization of previous expression.

A brief description of the prediction process after localization (classification) can be given as follows:

To compute the prediction of  $f_0$  we first expand both predictor and response on the same wavelet basis  $\psi^x$  and  $\psi^y$  on  $[a, b]$  and  $[c, d]$  time intervals respectively. Let  $A^0 = \{a_j^0\}_{j \in \mathbb{Z}}$  and  $B^0 = \{b_k^0\}_{k \in \mathbb{Z}}$  be the matrices of wavelet coefficients of an expansion.  $B^0$  is used only for comparing the results of prediction with the real values of trajectory. Then, we can compute the matrix  $K$  using expression above and calculate predicted wavelet coefficients using  $\hat{k}_{ij}$  and  $A^0$ :

$$\hat{B}^0 = \hat{\beta} \times A^0 \quad (47)$$

The obtained results are now the predicted wavelet coefficients for the second part of trajectory  $f_0$  on  $[c, d]$  time interval. Finally, inverse wavelet transform and adding  $\alpha$  gives the complete predicted positions of trajectory  $f_0$  on  $[c, d]$  time interval:

$$\{\hat{f}_0(s_i)\}_{s_i \in [c, d]} \quad (48)$$

We now have predicted the aircraft trajectory and p we can compare it with the real one on the experimental data set. One method we use is computing the relative prediction error (RPE), i.e. computing the ratio of the norm of the differences

between the predicted and the real trajectories by the variation of the given class. Which gives the following formula:

$$RPE_{f_0} = \frac{||\hat{f}_0 - f_0||}{\sqrt{1/(N-1) \sum_{i=1}^N ||f_i - \bar{f}||^2}} \quad (49)$$

## 6. Simulation Results

Our approach may be resumed as follows:

1. First of all, the right data set has to be chosen. The data set used in our research is one year air traffic observations over France. Moreover the observations have been cleaned from outliers and consist of only regular commercial IFR flights, i.e. all the VFR flights, military flights and others had been removed from the data set using operational ATM software. Some issues can appear in data processing corresponding to the commercial flights, like business jets that are flying on the same routes and the same altitude. But we do not pay attention on such kind of flight due to their variability, low frequency and variation from standard regular flights. Thus, in our work we use only the data that contains only standard regular commercial flights.
2. Then, all trajectories were classified into classes (or 'tubes') according to the origin/destination pairs (O/D pair). Before starting to build a regression model, we determine to which class belongs the trajectory to be predicted. In this paper we show several examples of predicted trajectories belonging to Paris-Toulouse, Paris-Marseille and Paris-Nice O/D pairs.
3. Next step is for the prediction itself. Let us denote by the "actual trajectory" the trajectory to be predicted. We now select the  $k$  nearest neighbors to the

'actual' trajectory in terms of shape and time from O/D pair corresponding to the 'actual' trajectory. Such an approach allows defining the trajectories with similar characteristics. The number  $k$  of nearest neighbors was chosen using leave-one-out cross validation procedure, described in chapter 3 above.

4. Now, the "actual trajectory together with its  $k$  nearest neighbors have to be divided into two parts, Predictor and Response, by time. After, both parts Predictor and Response of the selected  $k$  nearest neighbors are taken as a learning data set to construct a model, which is a linear functional regression model. Finally, Predictor part of an 'actual' trajectory has been used as an entry to the model. Predicted positions are then compared with Response part of the "actual" trajectory. Such comparisons can be seen from the figures 15, 16, 17 and 18.

To evaluate the efficiency of the algorithm a series of observations of real flights were used, with a different number of aircraft. The data set used in this work is an ATC radar observations over France during one year (see figure 14) and has original form of coordinates in Latitude, Longitude and Altitude. But, due to the small size of observing surface, we transformed Latitude and Longitude to standard 2D space ( $X \times Y$  plane), using stereographic projection with the reference point located at  $47^\circ 0' 0'' N, 0^\circ 0' 0'' E$ . We use  $Z$  coordinate instead of altitude, keeping the same units (flight level), to be more convenient in notations with  $X \times Y$  plane. As for the origin-destination pairs, Toulouse-Blagnac and Paris, Paris and Marseille, Paris and Nice airports had been chosen from the data set. The data set consists in more than 2.5 million aircrafts trajectories, which form the table with 12 columns and more than  $6 \times 10^7$  lines. Here the first and

second columns indicate date and time, third column corresponds to the flight identification number, the next six columns are the  $X$ ,  $Y$  and  $Z$  coordinates of an aircraft and projected speeds ( $VX$ ,  $VY$  and  $VZ$ ) of the aircraft on each of the coordinate axes, and the last 3 columns are aircraft type, origin airport and destination airport, respectively. The time interval of observations is not equidistant with increments from 3 to 30 seconds.

Place figure 14 about one day air traffic over France here.

Below we show 3 random examples from the list of predicted trajectories. They are figures 15, 16 and 17, where  $X$ ,  $Y$  and  $Z$  coordinates are shown for each example. The prediction bounds are equal for all experiments (approximately 25 minutes) with a starting prediction point at the 256th second on time axis. All the experiments that are examined involve approximately 4500 aircraft each, flying between Paris Orly and Nice, Marseille and Toulouse airports respectively. These results demonstrate that the algorithm based on wavelet decomposition has been successfully used to extract all the available information from the data.

The obtained results are convincing: most prediction errors can be explained by ATC controllers actions that make the descending aircraft keep constant altitude for avoiding a conflicting aircraft. For example, if a controller observes a conflict situation between landing aircraft and an aircraft on take-off, he directs the landing aircraft to keep a constant altitude to avoid the conflict. For example Figure 16 showss bigger prediction errors on the altitude prediction between 300th and approximately 350th seconds at  $Z$  coordinate. By essence, it is not possible to forecast such actions that are depending on the surrounding traffic.

Place figures 15, 16, 17 and 18 about predicted trajectories here.

Figure 18 demonstrate the predictions of altitude ( $Z$  coordinate) of an aircraft and its real position in a 10 minutes time horizon. There are two figures: the first one is the result of the method given at the third part of State-of-the-Art and the second figure shows the result of our approach. On both figures prediction starts at the point equals to 160th second. In fact, the difference between the real altitude position of an aircraft and the predicted (our approach) position at the last prediction point (760th second) is 72 ft. Vice-versa, the same difference in prediction with imprecise regression is 4000 ft. Such a difference in the results can be explained by the fact of generalization and over fitting of the model. It has been described in first section of State-of-the-Art that *Bias/Variance dilemma* dilemma has an influence on the quality the model and is also shown by this figure.

Relative prediction errors for 40 predicted aircraft trajectories are shown on figure 19. These errors were computed by equation 49, which is described at the end of the previous chapter. As it can be seen from the figure, the variation of the relative prediction errors from the standard deviation of the nearest neighbors is about 2% to 16%. Given results produced by this new approach show efficiency and ability to make prediction in 25 minutes time horizon.

Place figure 19 about relative prediction errors here.

## 7. Conclusion and Future Work

In this paper a novel trajectory prediction approach was studied, implemented and developed for Air Traffic Management (ATM) applications. The primary objective is to improve ATM capabilities and search for algorithmic methods that can use such a model efficiently to extract useful information. At the beginning of the paper, several existing methods with different approaches have been described.

These are standard multiple linear regression, an imprecise regression and neural networks based prediction.

The method presented here uses only previous radar tracks for a given origin-destination pair. A learning process has been used to adjust parameters. It begins from localization of the data set using  $k$  nearest neighbors and ends with solving the functional linear regression using wavelet decomposition in Sobolev space. Different wavelet bases were used to evaluate and to compare prediction errors. This method produces efficient results with high robustness. The validation of the proposed algorithm has been strengthening through extensive simulations, under varying scenarios and settings. The so-obtained algorithm, based on functional regression using wavelet decomposition has been compared to the first two existing methods presented at the beginning of the paper.

In the future step, the performance of the algorithm needs to be tested on a larger database, containing the flights over Europe. Moreover, prediction model will be developed by taking into account the weather variable with observations over the same period of time as the data set. Finally, the limit of this approach will be determined by increasing the prediction time interval. Future work also suggests development of the Conflict Detection algorithm based on aircraft trajectory prediction approach presented in this paper.



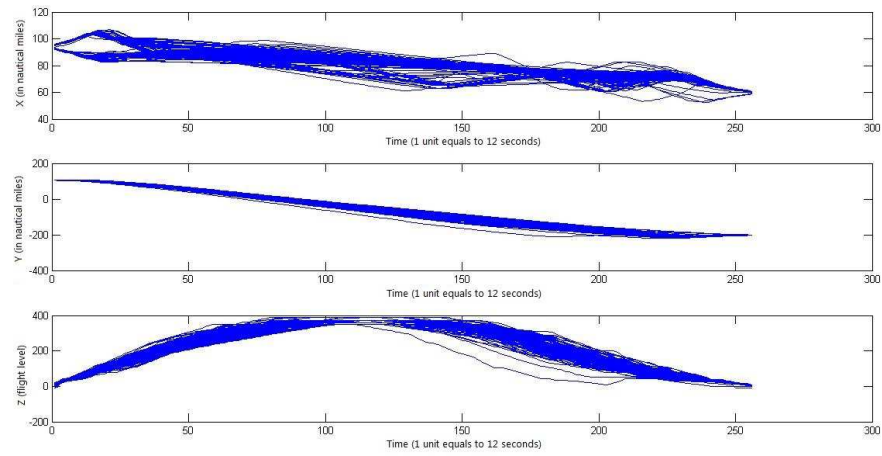


Figure 1: Training set for multiple linear regression.

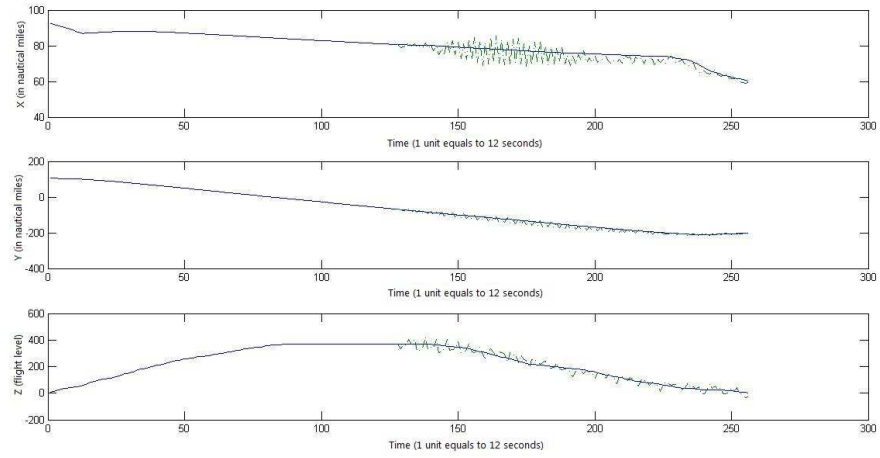


Figure 2: Altitude trajectory prediction (dashdot line) and the original path (line).

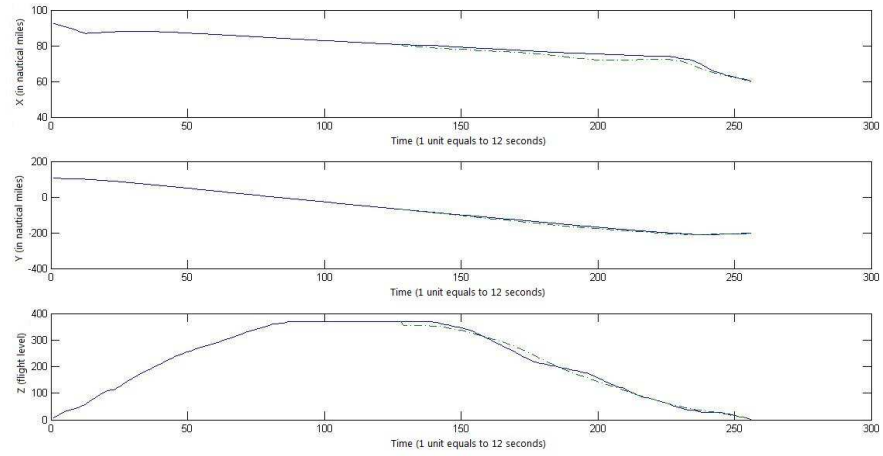


Figure 3: Predicted trajectory (dashdot line) and real trajectory (line).

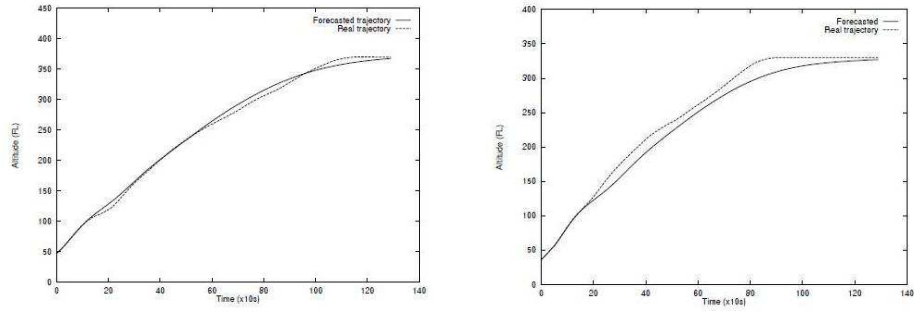


Figure 4: Example of a real trajectory and the corresponding prediction (left) and real trajectory not included in the learning base and corresponding prediction (right).

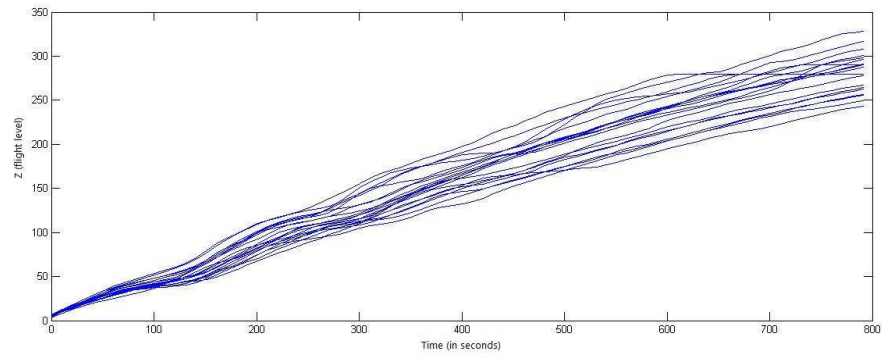


Figure 5: Example of altitude of 20 aircraft trajectories during take-off.

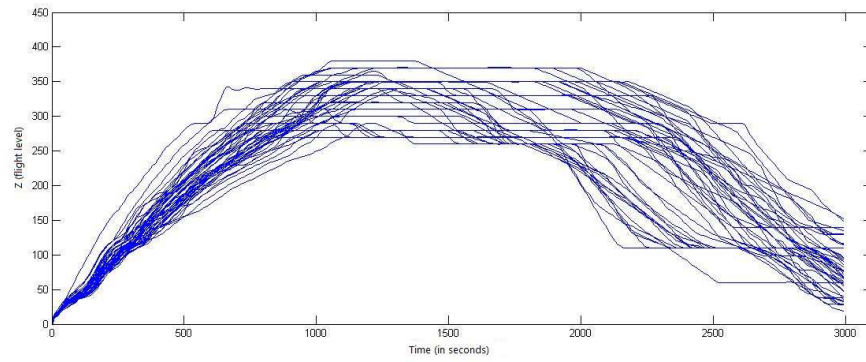


Figure 6: 23 aircraft trajectories (altitude) during take-off, cruising and landing.

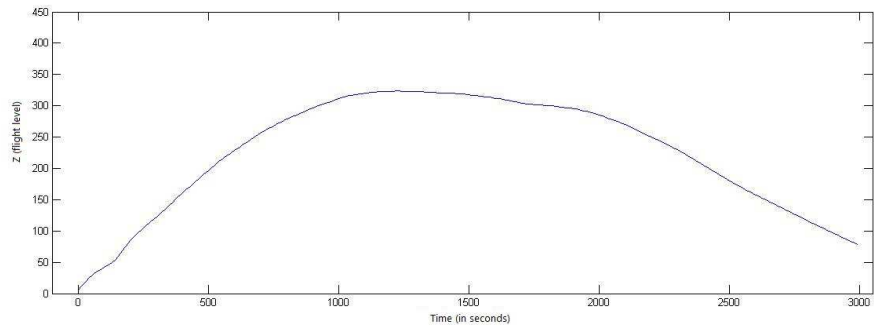


Figure 7: Mean of the 23 aircraft trajectories (altitude) of Figure 6

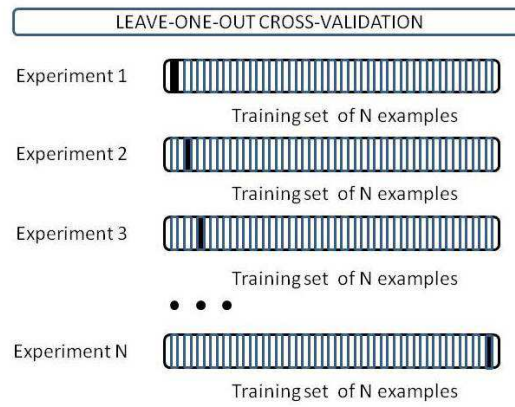


Figure 8: Leave-one-out cross-validation procedure.



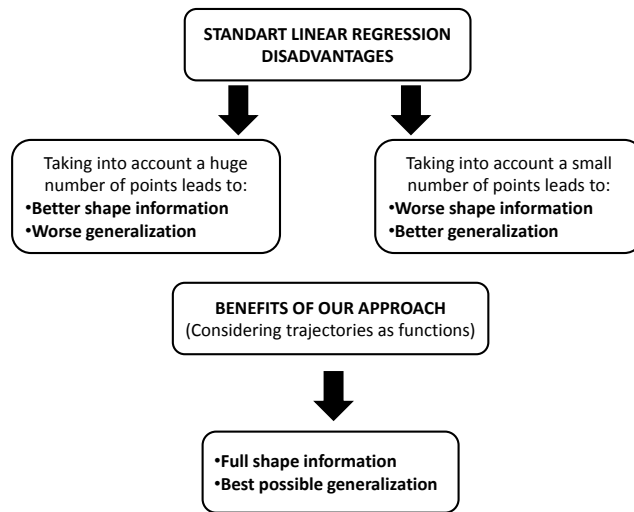


Figure 9: Benefits of considering trajectories as functions.

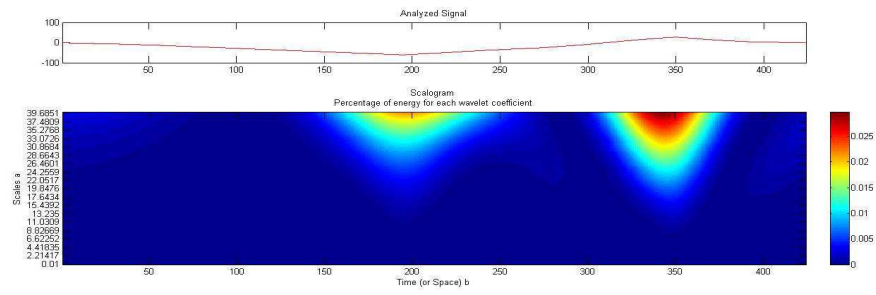


Figure 10: An aircraft's x coordinate and its CWT

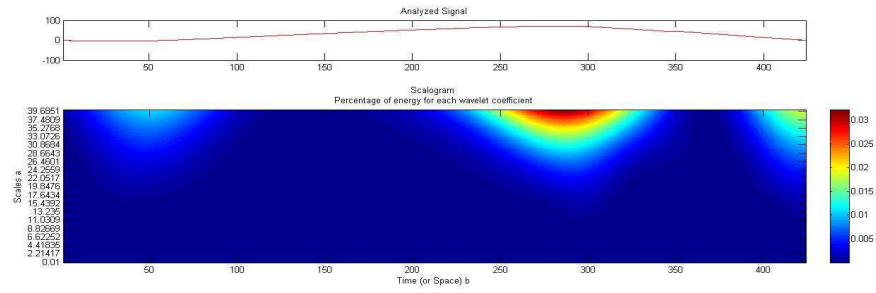


Figure 11: An aircraft's y coordinate and its CWT

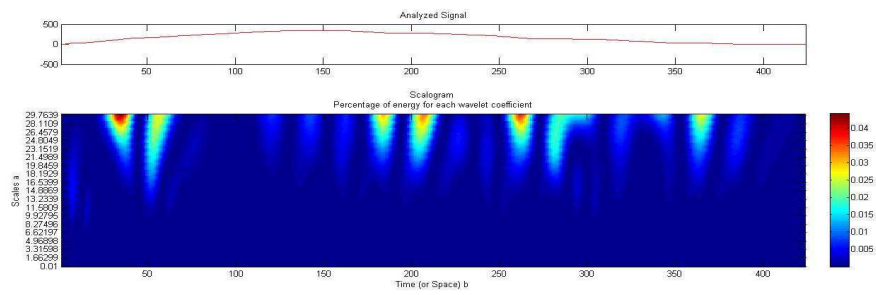


Figure 12: An aircraft's z coordinate and its CWT

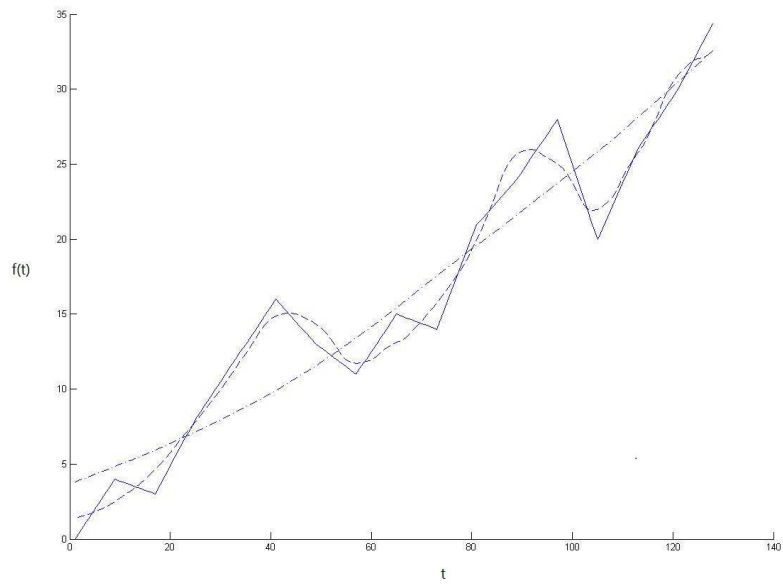


Figure 13: Smoothing and compressing an artificial function with wavelets.

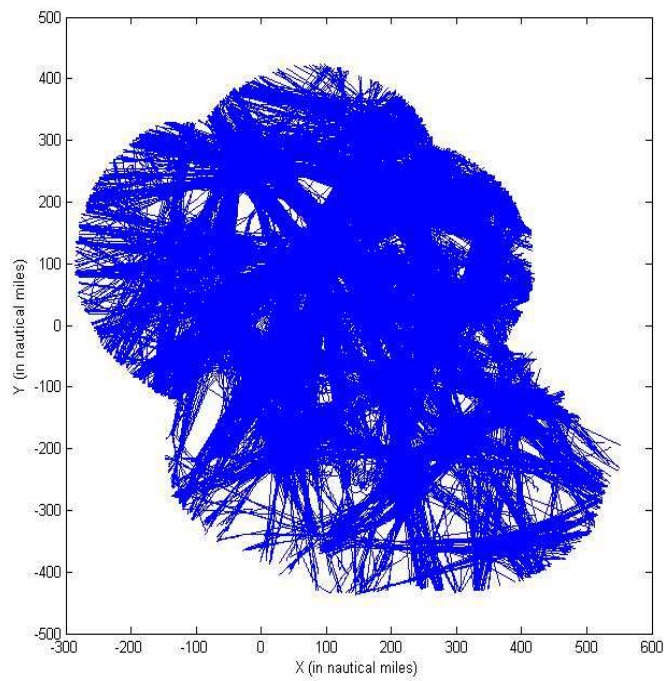


Figure 14:  $X \times Y$  plane of one day air traffic covered by ATC radars in France.

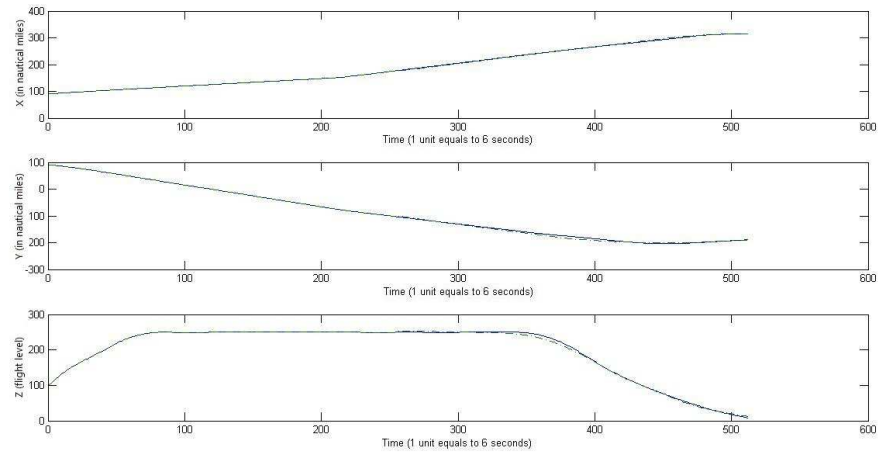


Figure 15: Real trajectory (line) and Predicted trajectory (dashdot line). Paris Orly - Nice.

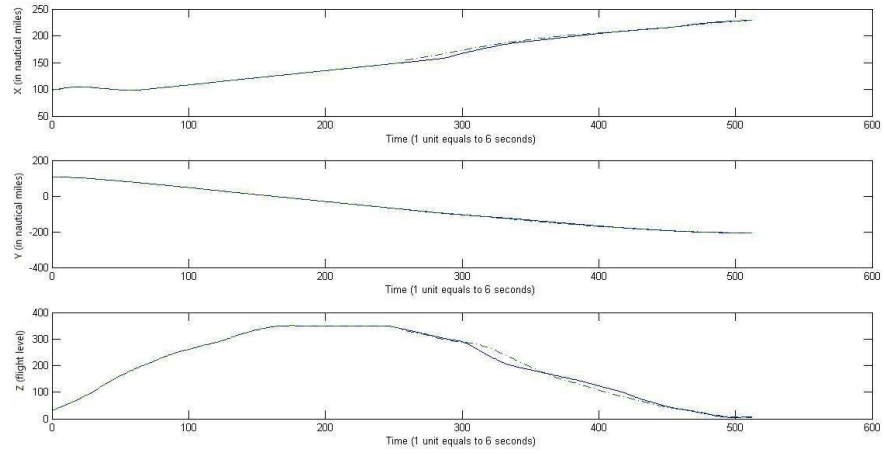


Figure 16: Real trajectory (line) and Predicted trajectory (dashdot line). Paris Orly - Marseille.



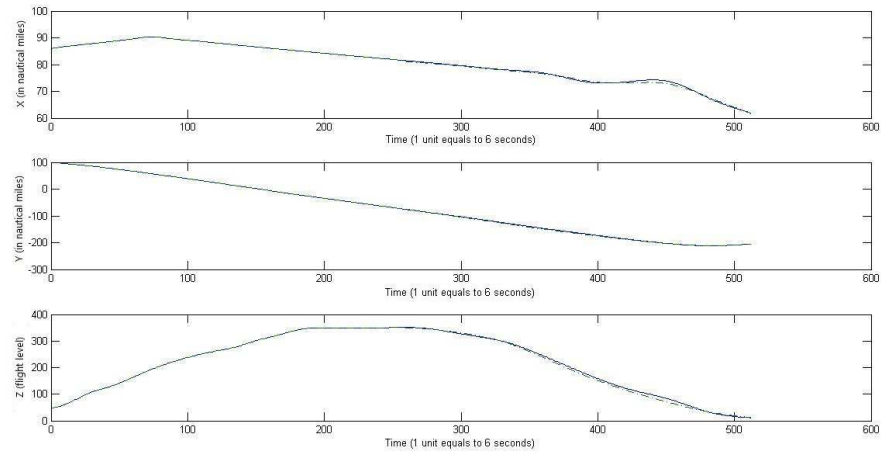


Figure 17: Real trajectory (line) and Predicted trajectory (dashdot line). Paris Orly - Toulouse

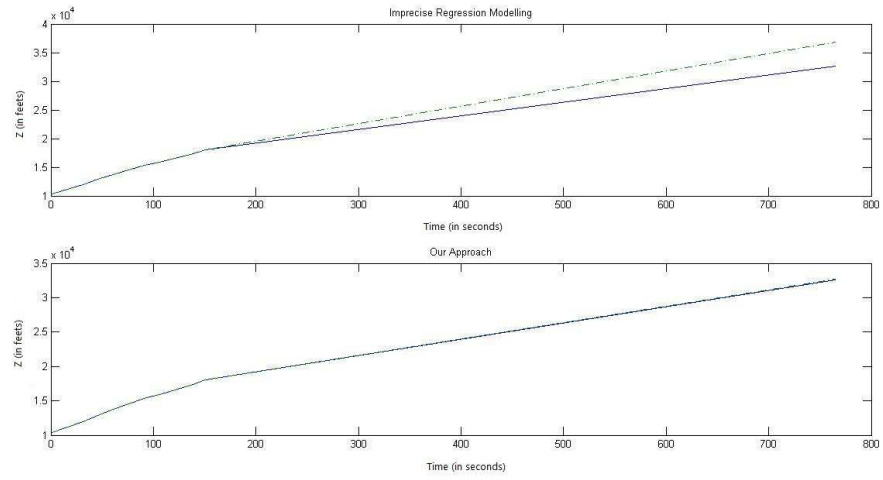


Figure 18: Real trajectory (line) and Predicted trajectory (dashdot line).

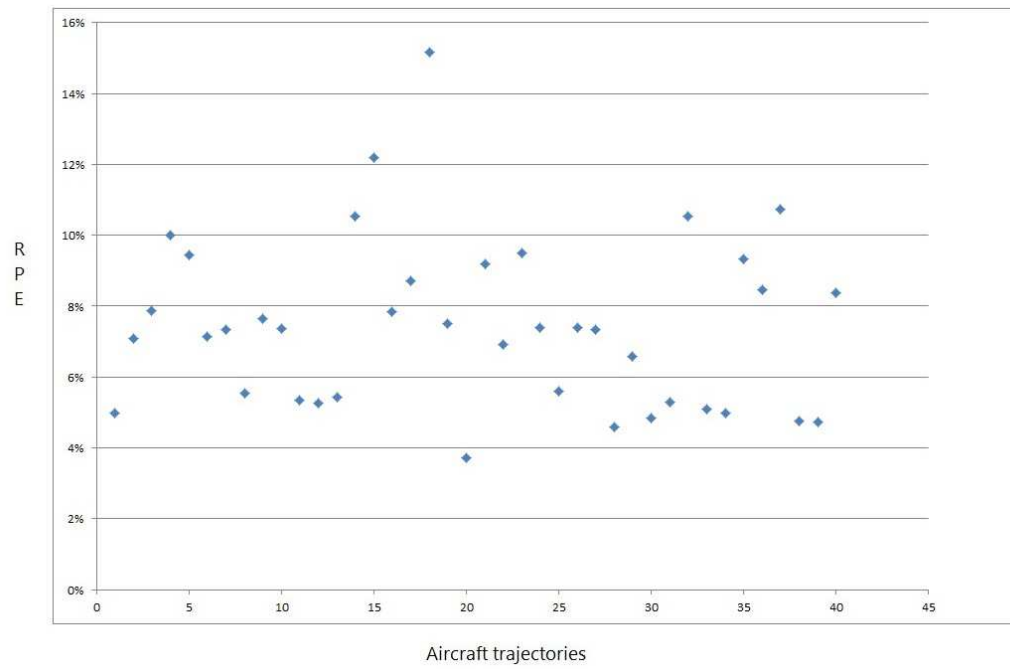


Figure 19: Relative prediction errors for 40 trajectories between Paris and Toulouse airports.

- A.Aguilera, F., M.Valderrama, 2005. *Functional and operational statistics: Estimation of Functional Regression Models for Functional Responses by Wavelet Approximation*.
- A.Baillo, A.Grane, 2007. *Local linear regression for functional predictor and scalar response*. Statistics and econometric series, pages 07-61 pp.
- A.Papoulis, S.Pillai, 2001. *Probability, random variables and stochastic processes*.
- B.F. Yao, H.G.Muller, J., 2005. *Functional linear regression analysis for longitudinal data*. Information theory, IEEE transaction on, 33.
- C.Gong, D.McNally, 2004. *a methodology for automated trajectory prediction analysis*. AIAA Guidance, Navigation, and Control Conference and Exhibit.
- C.G.S.Kirkpatrick, M.P.Vecchi, 1983. *Optimization by simulated annealing*. JSTOR, Science, New Series, Vol. 220, No. 4598. (May 13, 1983), pp. 671-680.
- Daubechies, I., 1990. *The wavelet transform, time-frequency localization and signal analysis*. Information theory, IEEE transactions on, 36:961-1005.
- Daubechies, I., 1992. *Ten lectures on wavelets*. Society for Industrial and Applied Mathematics.
- F.Ferraty, P.Vieu, 2006. *Nonparametric functional data analysis*.
- G.Lebanon, 2010. *Bias, variance and mse of estimators*.
- H.Choi, J., August 2010. *Continuous trajectory planning of mobile sensors for informative forecasting*. Automatica Volume 46, Issue 8, Pages 1266-1275.

- I.Gikhman, A.Skorokhod, 1996. *Introduction to the theory of random processes*.
- J.Demmel, 1997. *Applied numerical linear algebra*. Society for Industrial and Applied Mathematics.
- J.Doob, 1990. *Stochastic processes*.
- J.Ramsay, B.Silverman, 2005. *Functional data analysis*. Springer.
- M.Serrurier, Prade, H., 2004. *Imprecise regression and regression on fuzzy data, a preliminary discussion*. 2006 IEEE International Conference on Fuzzy Systems.
- R.Jia, J.Wang, D., 2003. *Compactly supported wavelet bases for Sobolev spaces*. Applied and computational Harmonic analysis 15, 224241.
- S.R.Doursat, E.Bienenstock, 1992. *Neural networks and the bias/variance dilemma*. Neural Computation, pages 4:1-58 pp.
- V.Meyer, 1992. *Wavelets and operators*.
- Y.L.Fablec, Y.Alliot, 1999. *Using neural networks to predict aircraft trajectories*.